Assembly Language and System Programming Lab8 Report

Group: 2

Names: 林柏廷(Brian)、蔡淵丞(Vincent)

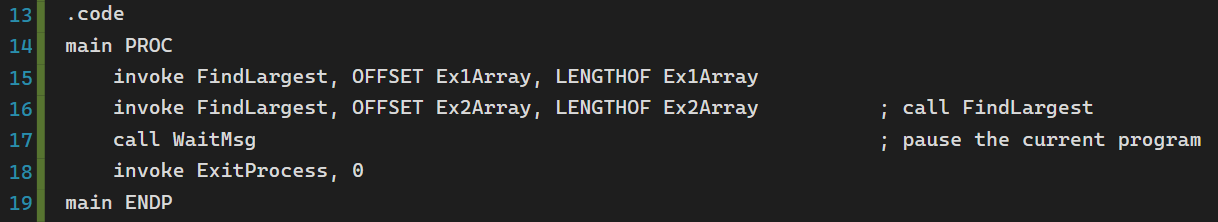
Student ID: 110502531、110502567

**Objective:**

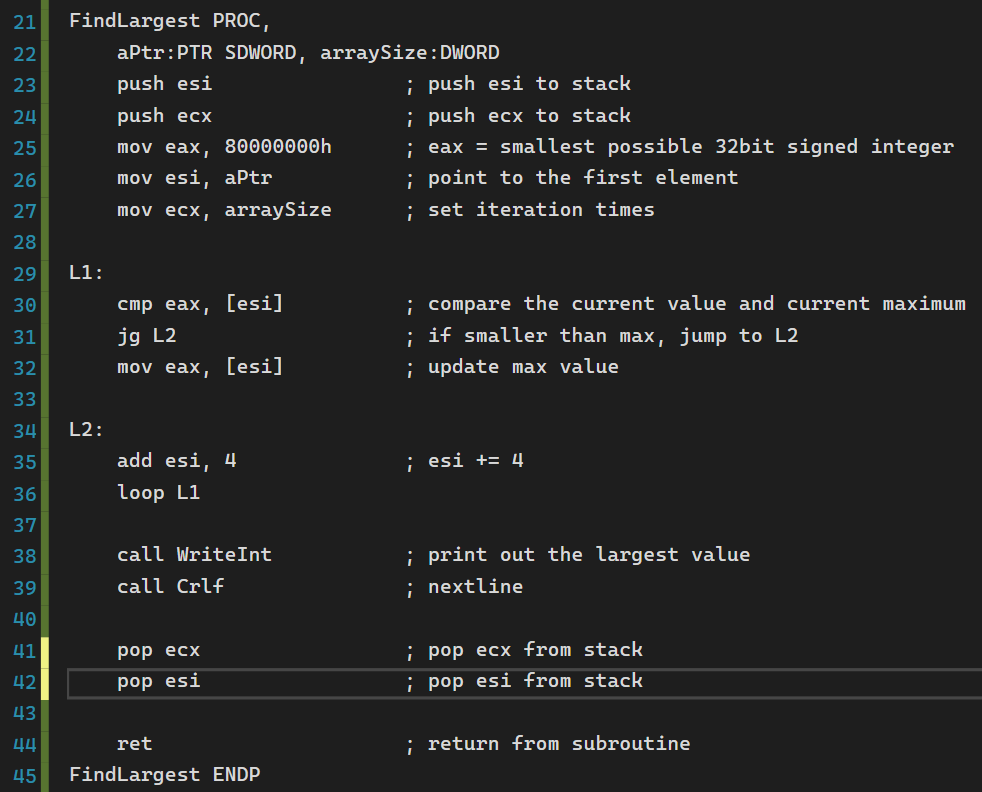
Find the largest number in an array and print it out on the screen.

**main PROC:**

Invoke the procedure twice with two different arrays and call WaitMsg, then invoke ExitProcess.



**FindLargest:**

****

**Line 22:**

Define two arguments aPtr and arraySize.



**Line 23-24, 41-42**

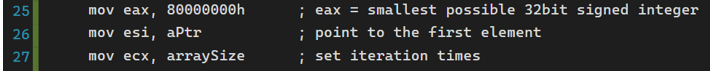
***Ecx*** and ***esi*** are used, therefore we use stack to store the value in order to avoid data loss.

******

******

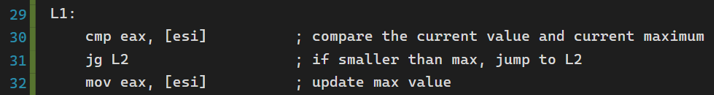
**Line 25-27:**

1. ***Eax*** is used to store the current maximum value, thus we move the smallest 32bit signed integer to ***eax.***
2. Move the pointer of the first element of the array to ***esi.***
3. Move the size of the array as the iteration time to ***ecx.***

****

**L1:**

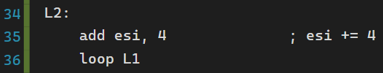
We compare the current value with the current maximum, if the current value is larger than or equal to the current maximum, then we replace the maximum with that value. If the opposite, then jump to L2.

****

**L2:**

Make ***esi*** point to the next element. (size of DWORD is 4)

Loop L1.

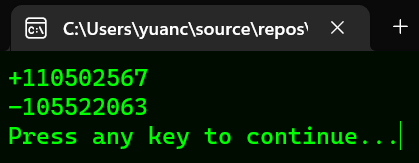


**Line 38-39:**

When the loop is done, print the maximum of the given array to the console.



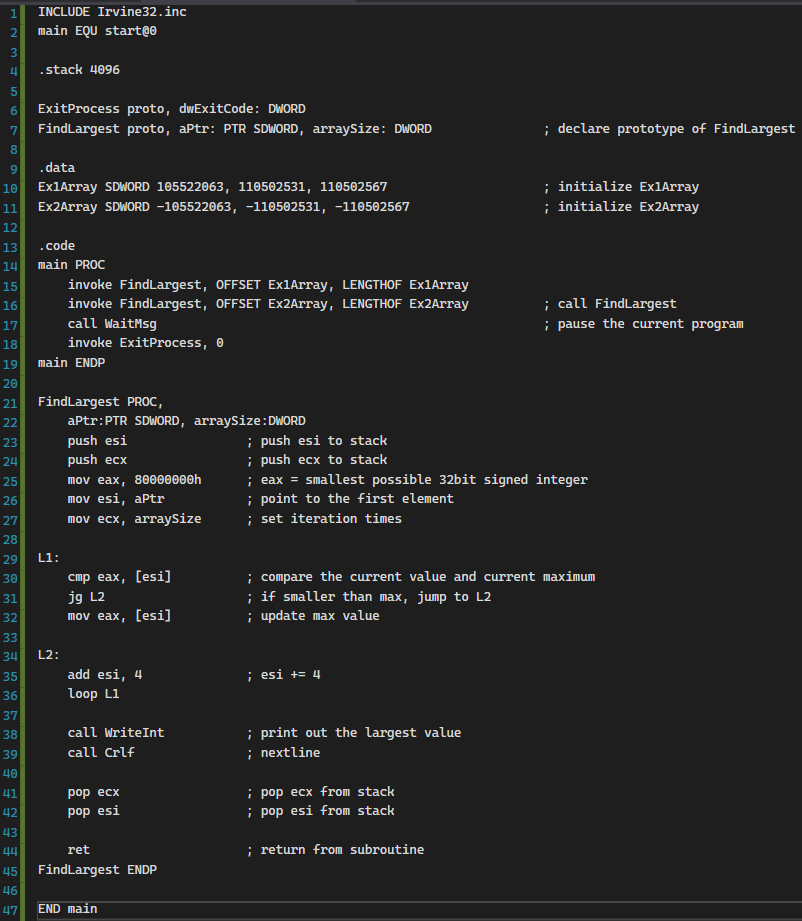
**Result:**

****

**Review:**

In this lab, we’ve learned two directives, invoke and proto. Invoke is a replacement for call that let us pass arguments to procedures. Proto appears at the top of the program, it creates a procedure prototype. The implementation occurs later. Every procedure called by invoke must have a prototype. This lab is significantly different from the previous ones since we’re provided with code segment.

**Full code:**

****